

AD-754 447

CONSTRUCTION OF AND EXPERIMENTATION WITH  
A FUNDUS REFLECTOMETER

William H. Bowie, et al

Technology, Incorporated

Prepared for:

Defense Nuclear Agency

November 1972

DISTRIBUTED BY:

**NTIS**

National Technical Information Service  
U. S. DEPARTMENT OF COMMERCE  
5285 Port Royal Road, Springfield Va. 22151



DNA 2930F

RESEARCH and

# TECHNOLOGY

DEVELOPMENT

INCORPORATED

LIFE SCIENCES DIVISION

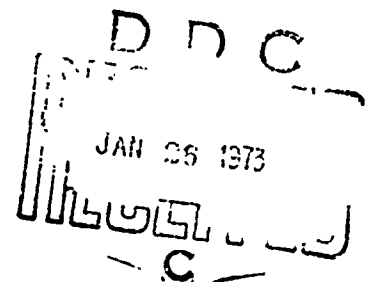
AD754447

CONSTRUCTION OF AND EXPERIMENTATION WITH A  
FUNDUS REFLECTOMETER

NOVEMBER 1972

FINAL REPORT

Reproduced by  
NATIONAL TECHNICAL  
INFORMATION SERVICE  
U S Department of Commerce  
Springfield VA 22151



Contract No. DASA 01-68-C-0149

Defense Nuclear Agency  
Washington, D.C.

Approved for public release -- Distribution Unlimited

381 N. NEW BRAUNFELS AVE. • SAN ANTONIO, TEXAS 78217

CONSTRUCTION OF AND EXPERIMENTATION  
WITH A FUNDUS REFLECTOMETER

FINAL REPORT

NOVEMBER 1972

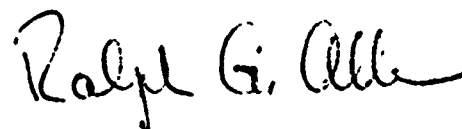
This work was supported by the Defense Nuclear Agency under NWER subtask  
MB167-01

William H. Bowie and Charles C. Stevens

Contract No. DASA 01-68-C-0149

Approved for public release; distribution unlimited.

APPROVED BY:



Ralph G. Allen, Ph.D.  
General Manager

The experimental work with volunteer human subjects discussed in this report was conducted in accordance with DOD Instruction 5030.29, dated 12 May 1964.

This research was conducted according to the principles enunciated in the "Guide for Laboratory Animal Facilities and Care," prepared by the National Academy of Sciences, National Research Council.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D		
<i>(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)</i>		
1. ORIGINATING ACTIVITY (Corporate author) Technology Incorporated, Life Sciences Division 8531 N. New Braunfels Avenue San Antonio, Texas 78217		2a. REPORT SECURITY CLASSIFICATION <b>UNCLASSIFIED</b>
		2b. GROUP ---
3. REPORT TITLE Construction of and Experimentation with a Fundus Reflectometer		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Final Report		
5. AUTHOR(S) (Last name, first name, initial) Bowie, William H. and Stevens, Charles C.		
6. REPORT DATE November 1972	7a. TOTAL NO. OF PAGES 453	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO. DASA01-68-C-0149	9a. ORIGINATOR'S REPORT NUMBER(S) None	
b. PROJECT NO. NWE R		
c. Subtask Code MB167	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d. Work Unit Code 01	DNA 2930F	
10. AVAILABILITY/LIMITATION NOTICES Approved for public release; distribution unlimited		
11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Director Defense Nuclear Agency Washington, D. C. 20305	
13. ABSTRACT <p>Efforts during the past year have been directed primarily to the collection and analysis of fundus reflectometry data from human and animal sources with particular emphasis on reflectance changes in the human fovea following both brief and extended photolysis. Studies of the retinal periphery and fovea of the rhesus monkey were performed at intensities above and below the threshold for retinal burns. A five-component model of retinal photopigment kinetics was utilized in the selection of exposures for the human experiments. Concurrent psychophysical flashblindness experiments with identical exposures were performed to allow comparisons between the data from both sources.</p>		

DD FORM 1 JAN 64 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Retinal photopigments, fundus reflectometry, flashblindness						

## INSTRUCTIONS

1. **ORIGINATING ACTIVITY:** Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (*corporate author*) issuing the report.

2a. **REPORT SECURITY CLASSIFICATION:** Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. **GROUP:** Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. **REPORT TITLE:** Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. **DESCRIPTIVE NOTES:** If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. **AUTHOR(S):** Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. **REPORT DATE:** Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. **TOTAL NUMBER OF PAGES:** The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. **NUMBER OF REFERENCES:** Enter the total number of references cited in the report.

8a. **CONTRACT OR GRANT NUMBER:** If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. **PROJECT NUMBER:** Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. **ORIGINATOR'S REPORT NUMBER(S):** Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. **OTHER REPORT NUMBER(S):** If the report has been assigned any other report numbers (either by the originator or by the sponsor), also enter this number(s).

10. **AVAILABILITY/LIMITATION NOTICES:** Enter any limitations on further dissemination of the report, other than those

imposed by security classification, using standard statements such as:

- (1) "Qualified requesters may obtain copies of this report from DDC."
- (2) "Foreign announcement and dissemination of this report by DDC is not authorized."
- (3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through \_\_\_\_\_."
- (4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through \_\_\_\_\_."
- (5) "All distribution of this report is controlled. Qualified DDC users shall request through \_\_\_\_\_."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. **SUPPLEMENTARY NOTES:** Use for additional explanatory notes.

12. **SPONSORING MILITARY ACTIVITY:** Enter the name of the departmental project office or laboratory sponsoring (paying for) the research and development. Include address.

13. **ABSTRACT:** Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (TS), (S), (C), or (U).

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. **KEY WORDS:** Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

UNCLASSIFIED

Security Classification

## FOREWORD

This report was prepared by the Life Sciences Division of Technology Incorporated. The report covers the period 11 May 1971 to 11 May 1972. Lt. H. J. Mitchell and Lt. Col. J. C. Cable were the Contract Monitors. This effort was supported by the Defense Nuclear Agency under NWER, Subtask Code MB167, Work Unit Code 01.

Acknowledgement is made of the assistance provided by the Contract Monitors during the period of the study.

## ABSTRACT

Efforts during the past year have been directed primarily to the collection and analysis of fundus reflectometry data from human and animal sources with particular emphasis on reflectance changes in the human fovea following both brief and extended photolysis. Studies of the retinal periphery and fovea of the rhesus monkey were performed at intensities above and below the threshold for retinal burns. A five-component model of retinal photopigment kinetics was utilized in the selection of exposures for the human experiments. Concurrent psychophysical flashblindness experiments with identical exposures were performed to allow comparisons between the data from both sources.

## LIST OF FIGURES

		<u>PAGE</u>
Figure 1.	Block Diagram of the Fundus Reflectometer	3
Figure 2.	Optical Density Differences Calculated Using Equation 1 on Data Obtained with Brief (40 msec) Exposure	13
Figure 3.	Optical Density Differences Calculated Using Equation 2 on Data Obtained with Brief (40 msec) Exposure	14
Figure 4.	Optical Density Differences Calculated Using Equation 1 on Data Obtained with Extended (100 sec) Exposure	15
Figure 5.	Optical Density Differences Calculated Using Equation 2 on Data Obtained with Extended (100 sec) Exposure	16



## TABLE OF CONTENTS

<u>SECTION</u>	<u>PAGE</u>
1. INTRODUCTION	1
2. THE APPARATUS	2
2.1 The Optical System	2
2.2 The Electronic Controls	4
2.3 The Computer System	5
2.4 Basic Operational Characteristics	6
3. EXPERIMENTAL METHODS	8
3.1 Primate Experiments	8
3.2 Human Experiments	8
3.3 Control Experiments	10
4. RESULTS	11
4.1 Primate Experiments	11
4.2 Human and Artificial Fundus Experiments	11
5. REFERENCES	18
APPENDIX	A - 1

## 1. INTRODUCTION

The primary objectives of this program were to investigate the rates of regeneration of the visual photopigments of living human eyes during flash-blindness recovery, and to investigate the reflectances of foveal and peripheral regions of rhesus monkey retinae after exposure to intensities above and below the thresholds for retinal burns.

Results of experiments on the rhesus monkey foveae have been reported previously.

During this reporting period, the primary emphasis has been on the investigation of flashblindness recovery in human subjects for brief and extended retinal illuminances. Results of these experiments are presented in this report, as are the results of investigation of the periphery of rhesus monkey retinae exposed to luminance intensities above and below retinal burn thresholds.

In addition, a more comprehensive program system for data acquisition and analysis was developed and installed. All data in this report were generated and analyzed using this new computer control system. A description of all components of this system is included as an appendix to this report.

## 2. THE APPARATUS

The evolution of the fundus reflectometer has been described in detail in previous reports. In its present final form, the fundus reflectometer is a computerized electro-optical system for the collection and reduction of retinal photopigment regeneration data. A block diagram of the system is presented in Figure 1.

### 2.1 The Optical System

There are three functional optical pathways in the fundus reflectometer. The first pathway, whose purpose is the production of the measuring beam and the capture of the reflections of this beam from the retina, consists of a tungsten-filament lamp, a series of lenses and aperture stops for the control of the beam as it enters the eye, a filter wheel containing a maximum of eight narrow-band-pass interference filters, and a collection mirror for directing the retinal reflections into the photomultiplier tube. The second pathway, which provides the bleaching light for the system, consists of a high-intensity xenon-arc light source, a system of lenses and mirrors for the control of the beam, a shutter, and a beam splitter to allow the superposition of this beam onto the measuring-beam pathway. The third pathway performs an ophthalmoscopic function allowing the experimenter to examine the subject's fundus for location and selection of a target area; it consists of a removable mirror placed at  $45^\circ$  to the optical axis between the

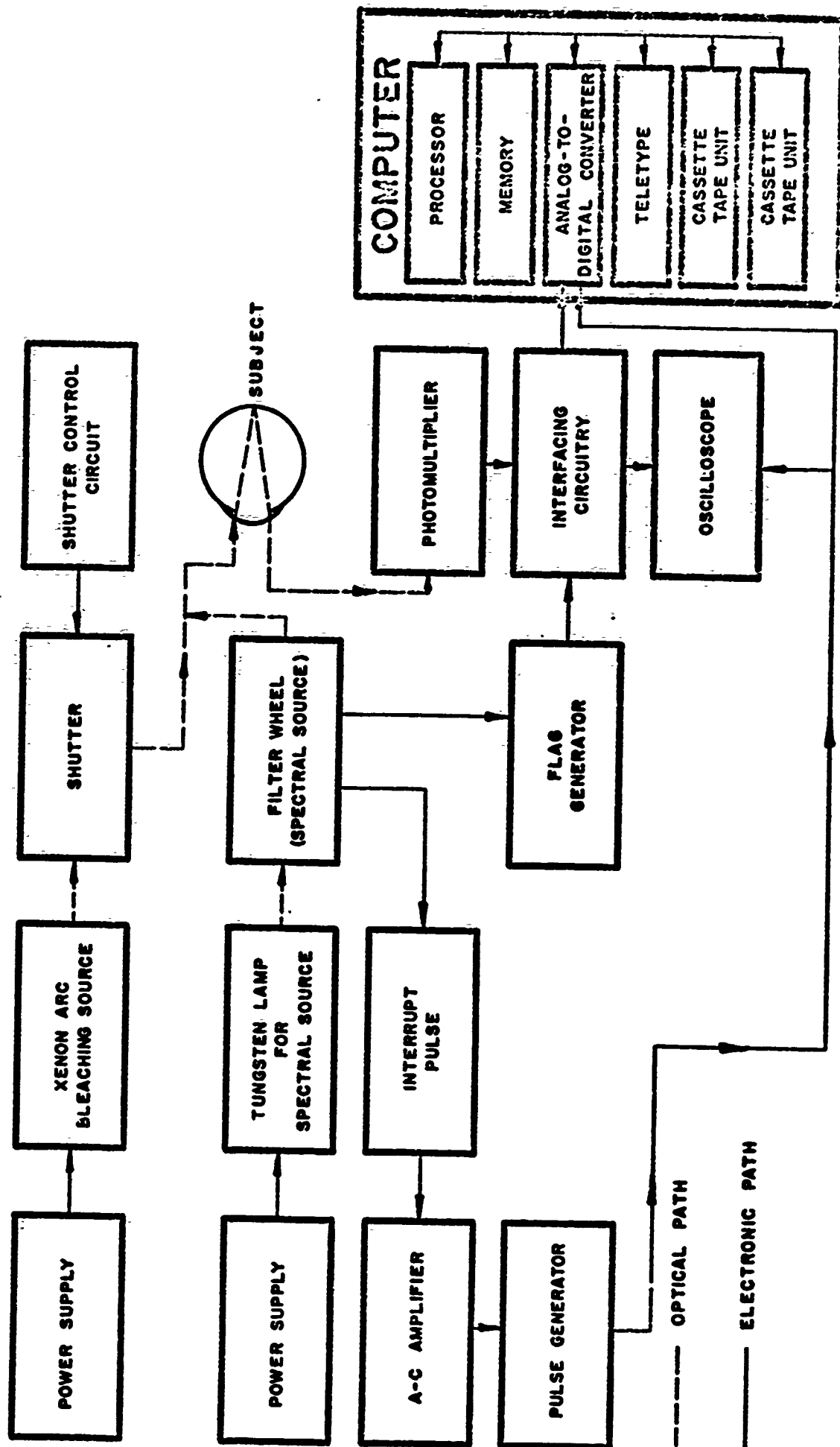


Figure 1. Block Diagram of the Fundus Reflectometer

collection mirror and the photomultiplier tube, and a series of lenses for magnification of the image.

## 2.2 The Electronic Controls

The electronic controls for the system provide amplification and noise-reduction of the signal from the photomultiplier tube. Pulse generation circuits for the control of data transmission between the optical system and the computer are also provided. The output of the photomultiplier is presented to one input line of an operational amplifier which masks the noise by means of an RC circuit and provides amplification of the signal. On the periphery of the filter wheel is mounted a cam which activates a microswitch once per revolution of the wheel; closure of this switch presents a signal to an alternate input line to the same amplifier. The net result is a large negative pulse, once per wheel revolution, in the normally positive output of the amplifier. Also mounted on the periphery of the filter wheel is an annular ring that is opaque except for a series of radially-inscribed slits. One slit is coincident with the cam and one with each filter mounted in the wheel. This ring is positioned between a small light bulb and a photodiode; as the wheel rotates, the slits allow light from the bulb to reach the photodiode. The output of this photodiode is connected to an operational amplifier and thence to a pulse generator which provides shaping and delay facilities for the pulses.

The output of the first amplifier is connected via a coaxial cable to the data input line of the analog-to-digital converter in the computer; it is this voltage that is treated as data by the system. The output of the pulse generator is connected via a coaxial cable to the interrupt input line of the converter. Each line is also connected to one trace of a dual-beam oscilloscope for real-time monitoring of the data.

The concurrence of a pulse on the interrupt line (called an "interrupt") and positive voltage on the data line is called a "data point"; the concurrence of an interrupt and negative voltage on the data line (whose source is assumed to be the cam on the periphery of the wheel) is called a "flag". The flag is used by the computer to maintain synchrony between subsequent cycles of the wheel by denoting the beginning of each cycle.

### 2.3 The Computer System

The computer system used to record and process data from the fundus reflectometer consists of an INTERDATA Model 3 processor with high-speed arithmetic and input/output option; 4,096 sixteen-bit words of core memory; ASR 33 teletype; two INTERTAPE cassette units with two tape drives each; and a single-channel analog-to-digital converter. The converter provides the interface between the fundus reflectometer proper and the computer. The teletype provides system control via its keyboard and hard-copy output through its printing

facilities. It is also used extensively in program development. The cassette units are used for program and data storage and retrieval.

#### 2.4 Basic Operational Characteristics

When all apparatus are readied for an experiment, the filter wheel is turning continuously and thus presenting a continuous train of data points and flags to the analog-to-digital converter. Since it is not possible for the computer to differentiate these events, provisions are built into the computer program to allow the operator to control the process by input of system control parameters in order to allow consistency of data and thus meaningful analysis of the results.

For all data included in this report, four narrow-band-pass interference filters (666 nm, 600 nm, 533 nm, and 466 nm) were mounted in the filter wheel, thus presenting four data points for each cycle of the wheel. The events resulting from one cycle of the wheel are referred to as a "scan"; in this context, therefore, a scan consists of a flag followed by four data points. The data were recorded in three logical blocks: one set of eight scans recorded with a dark slide preventing light from reaching the photomultiplier, in order to determine the quiescent voltage of the analog system; one set of eight scans recorded with the dark slide removed, allowing direct measurement of the dark-adapted state of the fundus; and twenty sets of three scans each recorded at thirty-second intervals after the flash (i. e., in each thirty second period, three scans

were recorded and then twenty-seven scans were skipped) to detect regeneration of the photopigments through reflectance changes. The set of data recorded on tape as a result of an experiment performed in the above manner is called a "standard run". At the completion of the experiment, any one of a number of processing programs may be loaded into memory at the discretion of the operator to effect analysis of the data, or a subsequent experiment may be initiated immediately. The ordinary procedure at the completion of an experiment was to select a simple data printout program to allow determination of data validity and proper recording. Since the data were recorded on tape, more complex processing was ordinarily delayed to allow more efficient utilization of the experiment time available.



### 3. EXPERIMENTAL METHODS

#### 3.1 Primate Experiments

Recovery in the peripheral retina of the rhesus monkey (macaca mulatta) following one-second retinal illuminances of either  $6.9 \times 10^6$ ,  $7.5 \times 10^8$ ,  $3.3 \times 10^9$ , or  $4.7 \times 10^9$  trolands was monitored using the fundus reflectometer. In addition, the photochemical effect of the fundus reflectometer measuring beam was examined.

To ensure immobility of the eye a head restraint device was constructed and mounted onto the same positioning bracket as the human bite-bars, permitting the same adjustments of head position as with humans. The eye was held in place by a system of sutures into the sclera. Corneal dehydration was prevented by manual closure of the eyelid. Little or no movement of the globe was observed during individual runs; however, it was often necessary to tighten the sutures between experiments.

At the conclusion of each experiment, the sutures were removed and the eye treated with a topical anesthetic and an antibiotic ointment. Careful observations during the recovery of the animals from the general anesthetic revealed no difficulty with the previously sutured eye.

#### 3.2 Human Experiments

Experiments were performed using human subjects for flash photolysis and extended photolysis of the foveal region. Upon employment, each subject received a thorough ophthalmological examination including

fundus photographs which were filed with the subject's personnel records. Dilation was performed prior to each day's experiments by the examining physician using Neo-syneprine hydrochloride 10%. The subject was then dark-adapted in the experiment room. Measurements were taken using the apparatus to determine the quiescent voltage of the system and the reflectance levels of the dark-adapted fundus.

For flash photolysis, the subject's eye was then exposed to a bleaching source of  $9.2 \times 10^7$  td for a period of 40 msec, giving an effective exposure of  $3.68 \times 10^6$  td-sec. For extended photolysis, the intensity of the source was  $9.4 \times 10^4$  td, the duration 100 sec, and the exposure  $9.4 \times 10^6$  td-sec. These exposure histories were chosen using the theoretical model of photopigment kinetics<sup>(1)</sup> to give approximately 70% bleach at the end of the exposure. Experiments in the concurrent flashblindness project were carried out to determine recovery times for identical illuminance-histories to allow comparison of data from the two studies. The measuring beam size, the flash beam size, and the position of the subject were all carefully coordinated to ensure that the area detected by the photomultiplier (approximately  $5^\circ - 6^\circ$  of visual angle) fell within both the area bleached and the area covered by the measuring beam (approximately  $10^\circ$  of visual angle). The subject was aided in fixation by the interposition of a small black dot in the center of the measuring beam. The photomultiplier tube was protected

from exposure to the bleaching source by a dark slide which was removed prior to recording of the regeneration data.

### 3.3 Control Experiments

Concurrent with the processing of human subjects, a series of experiments was made in which reflectances were measured from an artificial eye. This artificial eye was designed to approximate as closely as practicable the optical characteristics of the human eye. Mounted at one end of an aluminum housing was a meniscus lens of 1.7 cm focal length; at the other, an aluminum disk with a spherically concave inner surface to which was glued a piece of orange paper. The interior of the housing was lined with black paper to minimize internal reflections. Since the reflectances from the "fundus" (the orange paper) of this device do not change, these experiments provided a valuable tool in the determination of ambient noise levels of the system as they might affect the outputs of the various processing programs. Monitoring of the photomultiplier output during bleaching exposure with the dark slide in place failed to reveal any light transmission to the photomultiplier. Since no effect on the photomultiplier was observed as a direct result of the bleaching exposure, the inclusion of that exposure in the process of the artificial-fundus experiments was deemed unnecessary, and was omitted.

## 4. RESULTS

### 4.1 Primate Experiments

The  $4.7 \times 10^9$  td illuminance produced severe retinal burns which were readily observed ophthalmoscopically. Fundus reflectance increased immediately after exposure, and was still gradually increasing ten minutes later. As anticipated, normal dark adaptation or photopigment regeneration was not observed.

The three cases of lower illuminance exposures did not produce retinal burns, and photopigment regeneration was observed. In these cases, reflectance appeared to increase slowly for the first 100 sec, then to decrease slowly thereafter as anticipated. Although experimentation without the use of the bleaching exposure showed a similar rise in reflectance for the first 120 sec (assumed to be the bleaching effect of the measuring beam), verification and explanation of the phenomenon would require significantly more data.

### 4.2 Human and Artificial Fundus Experiments

Ten experiment runs were made using each of the two bleaching intensities. The data from each group of ten runs was used as input to a program which calculated optical density differences <sup>(2)</sup> for each run (See Appendix 1), and formed mean optical density differences over the ten runs.

The single set of eight baseline scans and each of the twenty sets of three post-flash scans were averaged to produce a single mean baseline scan and twenty mean post-flash scans respectively. The optical density differences were calculated on the basis of these mean scans to present single data values at the pre-flash point and at thirty-second intervals after the flash. These optical density differences were then averaged over the ten runs in the group to produce the results displayed and marked with circles in Figures 2 - 5. These figures also contain the results of the artificial fundus experiments (marked with triangles), analyzed and presented in the same manner as the human data. Since no bleaching light was used in the artificial fundus experiments, the data presented for flash photolysis and for extended photolysis of the artificial fundus are in fact the same data.

The optical density differences were calculated in two forms: difference between later post-flash scans and the scan immediately following the completion of bleaching exposure (Figures 2 and 4), and between the baseline and each post-flash scan (Figures 3 and 5). In all cases the data obtained at 666 nm were used as a reference point according to established fundus reflectometric procedure.<sup>(3, 4)</sup> Given output voltages  $V$  from the fundus reflectometer, and further given that  $V_i$  is a data point from the first post-flash scan,  $V_{ir}$  is the data point obtained at 666 nm in that scan,  $V_b$  is a data point from the baseline scan,  $V_{br}$  is the 666 nm point in that scan.  $V_p$  is a data point from a post-flash scan, and  $V_{pr}$  is the 666 nm point in that scan, the two formulae used to obtain optical density differences were

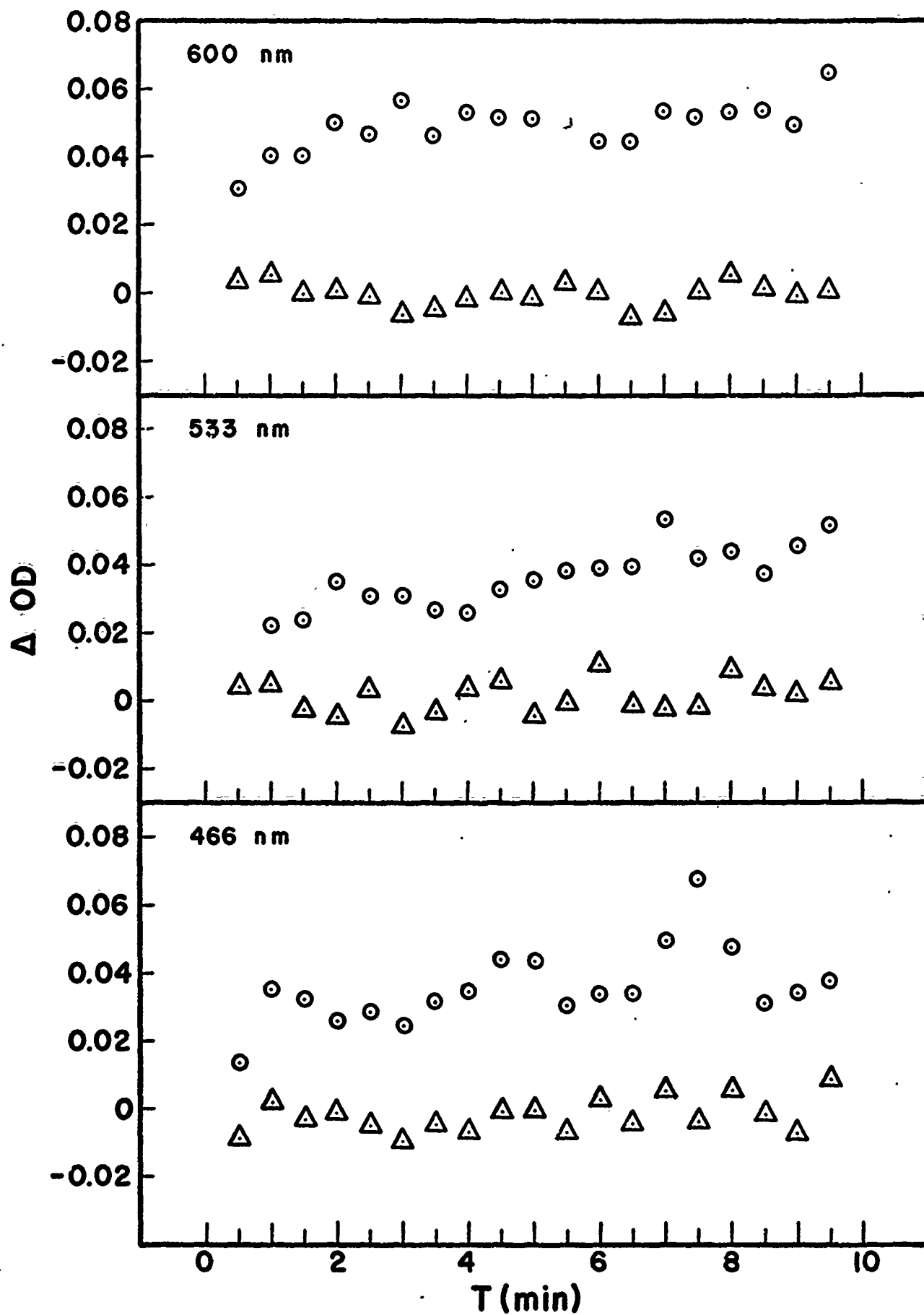


Figure 2. Optical Density Differences Calculated Using Equation 1 on Data Obtained with Brief (40 msec) Exposure

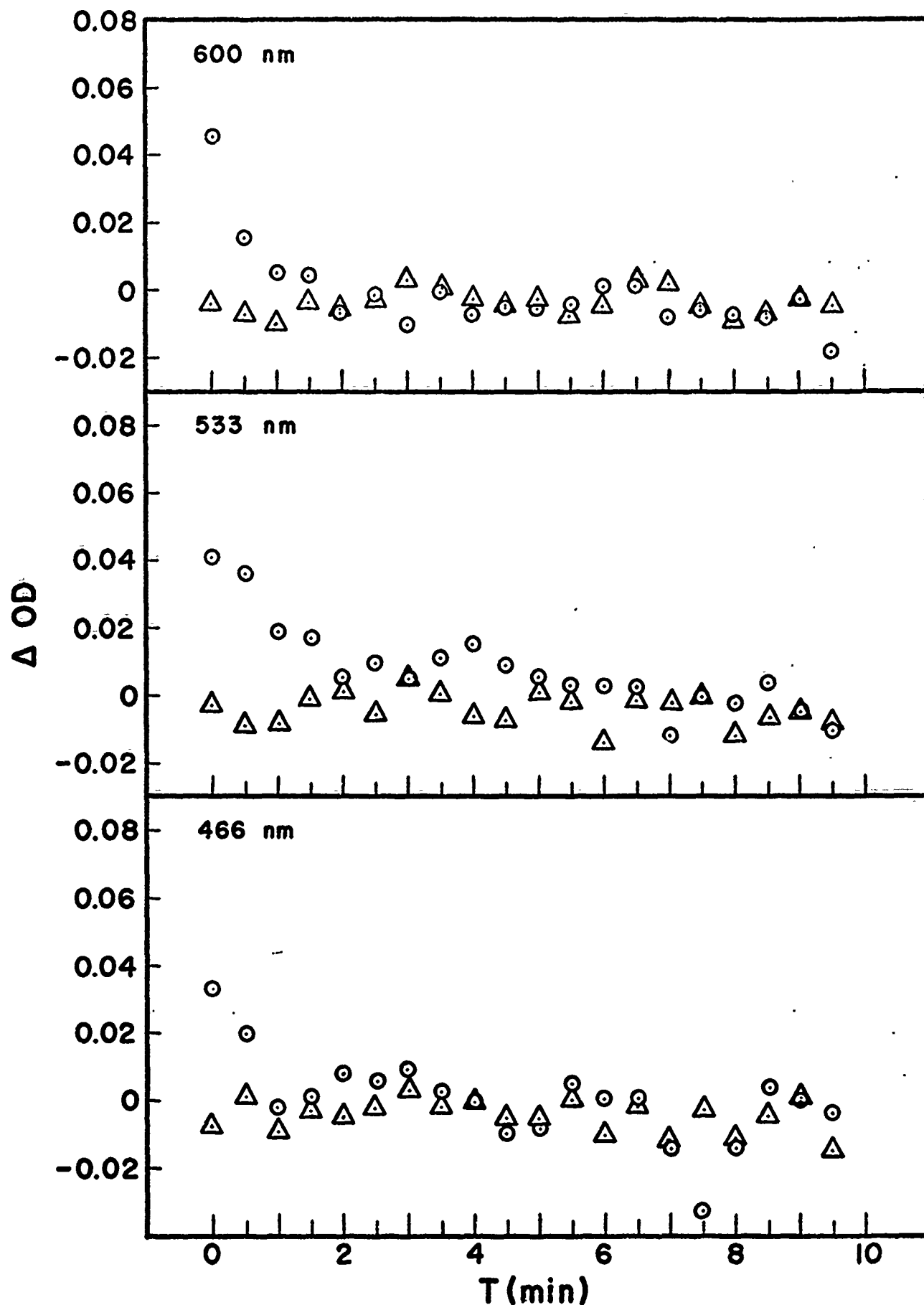


Figure 3. Optical Density Differences Calculated Using Equation 2 on Data Obtained with Brief (40 msec) Exposure

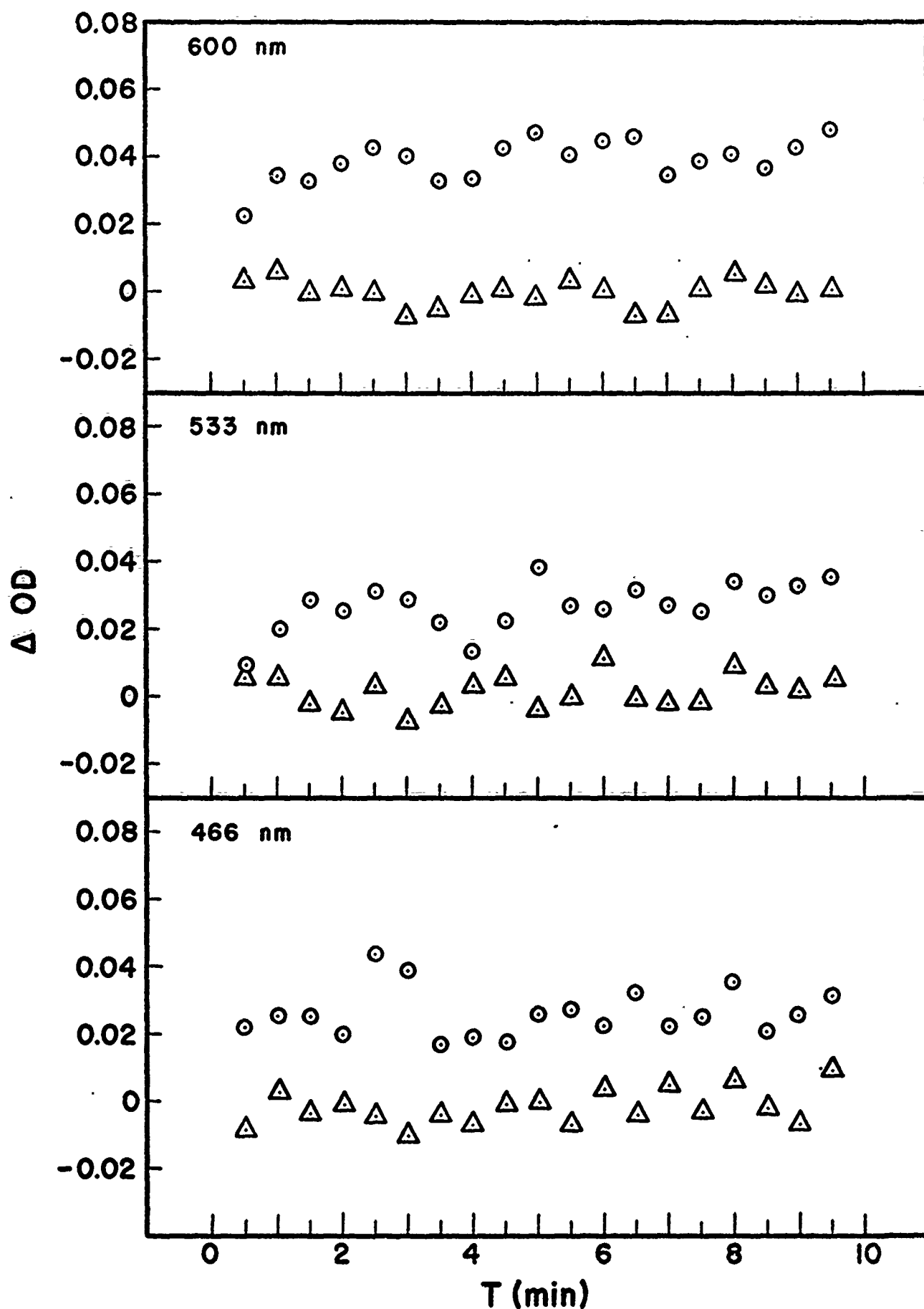


Figure 4. Optical Density Differences Calculated Using Equation 1 on Data Obtained with Extended (100 sec) Exposure



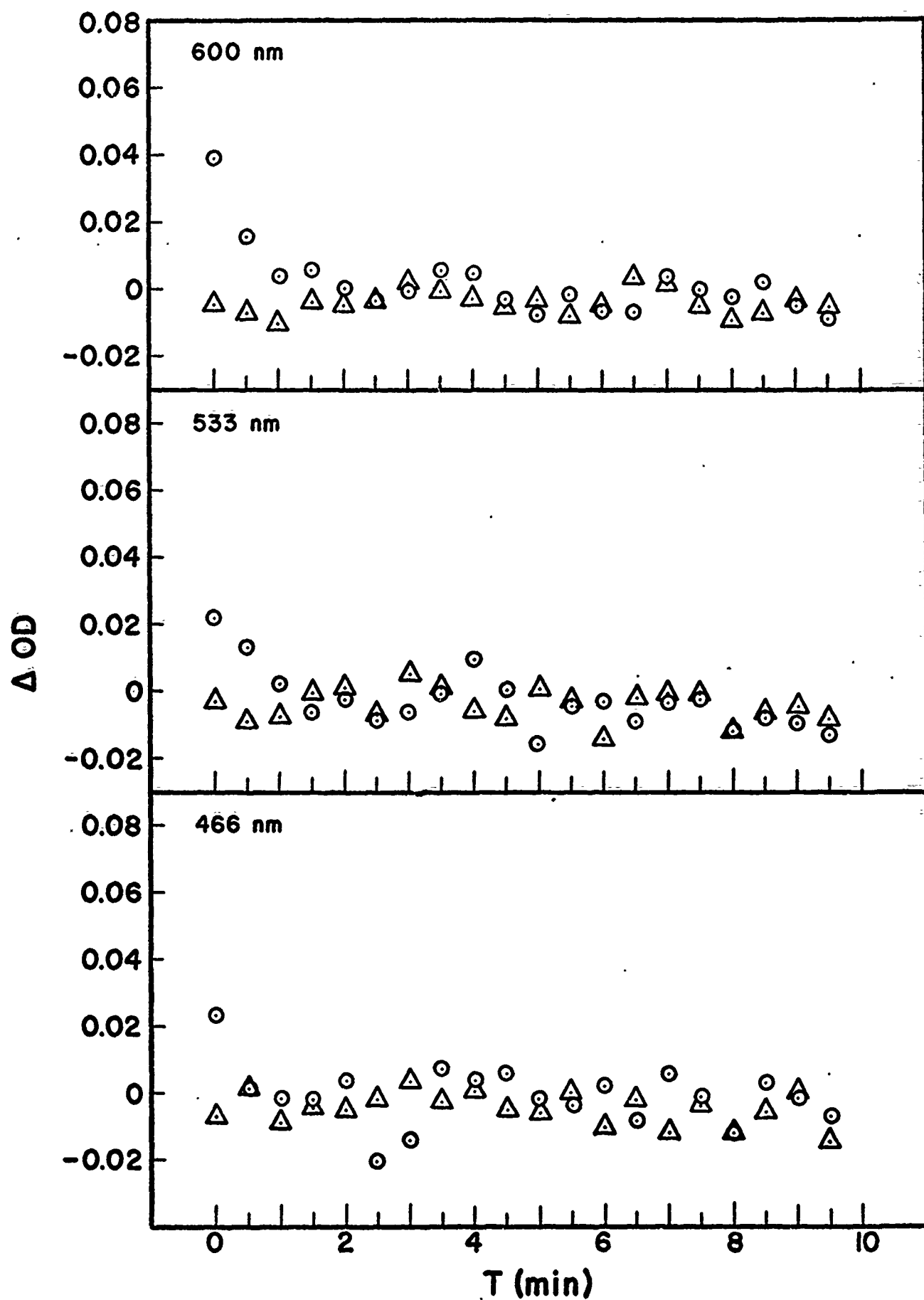


Figure 5. Optical Density Differences Calculated Using Equation 2 on Data Obtained with Extended (100 sec) Exposure

$$\Delta OD = \text{Log}_{10}((V_i/V_{ir})/(V_p/V_{pr})) \quad (\text{Eq. 1})$$

and

$$\Delta OD = \text{Log}_{10}((V//V_{pr})/(V_b/V_{br})) \quad (\text{Eq. 2})$$

respectively. <sup>(2)</sup> Figures 2 and 3 present data from the flash photolysis experiments, and Figures 4 and 5 data from extended photolysis.

Optical density differences calculated from the artificial fundus results shows the expected characteristics: they were essentially linear with time and remained very close to zero. The optical density differences calculated from the human experiment data also were as expected in that they showed a high reflectance immediately after the flash, decreasing slowly to pre-flash levels. The magnitude of the differences obtained with this system is somewhat lower than that of Weale <sup>(2)</sup> due in part to the impracticality of obtaining reflectance data in this system within the first few seconds after bleaching exposure. The data, however, are directly comparable and show that the system is correctly measuring reflectance changes in the human retina during the regeneration of photopigments.

## 5. REFERENCES

1. Mainster, M.A., White, T.J., and Stevens, C.C. Mathematical analysis of rhodopsin kinetics. *Vision Res.* 11, 435-447.
2. Weale, R.A. Photo-sensitive reactions in foveae of normal and cone-monochromatic observers. *Optica Acta* 6, 158-174.
3. Rushton, W.A.H. and Henry, G.H. Bleaching and regeneration of cone pigments in man. *Vision Res.* 8, 617-631.
4. Alpern, M., Mäaseidvaag, F., and Ohba, N. The kinetics of cone visual pigments in man. *Vision Res.* 11, 539-549.
5. Hart, J. F., Cheney, F. W., et al., Computer Approximations. John Wiley & Sons, Inc. New York, 1968.

APPENDIX

Data Acquisition and Analysis Programs

*A*

During the period covered by this contract, certain enhancements were made to the computer system which is devoted to the collection and analysis of fundus reflectometry data. The addition of cassette tape units to the computer has allowed the development of a data acquisition and analysis system which permits the separation of acquisition and processing functions into separate programs, as well as the offline storage of fundus reflectometry data for subsequent processing.

The system consists of a generalized analog-to-digital converter driver (whose basic purpose is the initial reading of fundus reflectometry data from the apparatus and the writing of these data on cassette) and a number of analysis and processing programs designed to accept the tape output of the driver. These programs are kept in machine-readable form on cassette, and are loaded into memory through a rudimentary operating system which positions the program tape and then loads the selected program using the standard system bootstrap loader. The set of all programs designed to run under this environment is called the Fundus Reflectometry Control System, or FRCS. The analysis programs currently operational under FRCS include items ranging in complexity from raw voltage printout programs for archival and diagnostic purposes to optical density difference calculators that accept multiple sets of like data as input.

A detailed description of the individual programs that make up the system follows. This description is designed to provide both programming and operating conventions relating to the system.

## FUNDUS REFLECTOMETRY CONTROL SYSTEM (FRCS)

This discussion assumes that the reader is familiar with the INTERDATA computer system and associated software. The INTERDATA 50 sequence, General Loader, and assembler are referenced in this discussion; the first two are explained more fully in INTERDATA publication 29-231, and the third in 29-230. INTERDATA standard notation is used throughout. The INTERDATA floating-point package is described in 29-172; any other information on the system may be found in 29-004R02.

Each element in the FRCS is written in INTERDATA Assembler Language, with relocatability specified. Each main program references a routine named MONITR, which must be loaded with it; MONITR in turn references PRTLIN. MONITR is always loaded in memory locations x'80' to x'FF', and the main program always follows MONITR such that the first executable instruction in the main program is located at x'100'. PRTLIN may be loaded anywhere in the sequence of subprogram loads following the mainline subprogram and preceding any open-ended storage definition subprograms.

The process of creating a system residence tape consists in essence of loading the subprograms into memory to form an executable program unit, then writing this program unit onto cassette. The loading of the subprograms is accomplished using the INTERDATA General Loader, and the creation of the program tape is performed using a machine-language routine written expressly for that purpose. When an executable program unit is written

on tape, both the General Loader and the output routine are included, so that they are available whenever a FRCS program is resident; however, some of the programs have been structured such that during processing the memory locations occupied by these routines may be overwritten. They are available, then, only immediately after program loading and before execution begins. The General Loader occupies memory locations x'1A00'-x'1FC3' and the output routine locations x'1FC4'-x'1FFF'. The output routine creates a core-image record on cassette which is to be loaded using the INTERDATA 50 sequence with location x'5A' altered to contain x'1FFF' instead of x'CF'. Each record is preceded by a file gap to allow high-speed tape positioning by MONITR, and comprises a core image of locations x'80' to x'1FFF'. As indicated above, each such record always contains MONITR in locations x'80' to x'FF', the General Loader in locations x'1A00' to x'1FC3', the cassette output routine in locations x'1FC4' to x'1FFF', and PRTLIN loaded via the General Loader and linked to MONITR at some point in memory between the end of the mainline program and the beginning of the General Loader.

MONITR is the "operatingsystem"; it is responsible for positioning the system residence tape properly when a program is selected through the low-order seven data keys on the system console, and transferring control to the INTERDATA 50 sequence for the actual loading process. At the completion of loading, MONITR checks to ensure that the program loaded was in fact the program selected, and if so transfers control to x'100', by convention the first executable location of the mainline subprogram.

MONITR may be accessed in one of two ways: by the program and by the operator. Access by the program is accomplished by branching to a defined entry point in the subprogram which causes the printing of the message "SELECT PROGRAM" and the halting of the processor, at which point the operator places the program number in hexadecimal in the low-order seven data keys and depresses execute; access by the operator (which dispenses with the message and halt) is accomplished by setting the current location counter in the PSW to x'80' and then entering the program number in the data keys as above. At the completion of either of these processes, if the program selected is resident, it is initiated at its first location (x'100'); if not, the program tape is rewound, and the proper program is located and loaded. This is accomplished whether entry to MONITR was manual or automatic. Restarting of a program known to be resident when MONITR is not in control is accomplished by a manual branch to x'100' which bypasses MONITR completely.

The program number which the operator uses to select the program has two meanings. It is used by MONITR as the number of files to skip on the system residence tape before loading, and it is then checked by MONITR at the completion of loading to ensure that the loading process transpired correctly. Accordingly, care should be taken that the program number and its location on tape be identical when the program unit is written onto the tape if the program unit is to be loaded using MONITR. If this precaution is not observed and loading is attempted through MONITR,



the processor will hang with the "USE" light lit on the residence drive and the program tape at end-of-tape.

The writing of a mainline subprogram and subsequent cataloging of the program unit on the system residence tape of FRCS is thus subject to several constraints:

- (1) The mainline subprogram, and all subprograms to be loaded with it, must be relocatable;
- (2) The mainline subprogram should declare as EXTRN the label MONITR, to which it branches upon job termination and it should likewise declare as ENTRY the label PROGNO, which should specify an integer halfword containing the file number that the executable program unit is to occupy when it is written on the program tape;
- (3) When the components of the executable program unit are loaded into memory using the General Loader, the following order should be observed: MONITR, mainline subprogram, ancillary subprograms, and open-ended storage definition subprograms; and
- (4) Among the ancillary routines loaded should be PRTLIN which is used by MONITR to type the end-of-job message.

## PROGRAM ABSTRACTS

Program 0, the program which reads the analog-to-digital converter and places the data on cassette, is a generalized program designed for flexibility. A standardized subset of the possible outputs of this program, called a standard run, conforms to certain restrictions. A standard run consists of three blocks of data: the first, containing one pattern repetition, is to be treated as a reading of the quiescent voltage of the system (called the bias voltage in later discussions), obtained with a dark slide over the photomultiplier tube; the second, containing one pattern repetition, is treated as a reading of the dark-adapted state of the fundus (referred to as the baseline); and the third, usually with multiple pattern repetitions, is treated as a reading of the bleached fundus and its recovery to the baseline state (called the post-flash data). "Block" and "pattern" are defined in the description of Program 0.

The majority of the processing programs have three characteristics in common: they are loaded by MONITR; they expect a single standard run as input; they exit through MONITR when processing of that run is complete. A program unit of this description is referred to as a standard processing program; all processing programs are assumed to be standard unless otherwise specified in their abstracts. Program 0, of course, is not considered a processing program at all in this context.

PROGRAM 0: Analog-to-Digital Conversion and Tape Output.

Label: FUNDAD. Entry for first instruction: FUNDAD.

Other entries: PROGNO, TAPE. Extns: PRTLIN,  
TAPIN, TAPOUT, ENDFIL, MONITR, DTOB, BTOD,  
TOTAP, INADC. Loading order MONITR, FUNDAD,  
PRTLIN, TAPIN, TAPOUT, BTOD, DTOB, STORES.

Storage for mainline subprogram: x'A60'.

For the purposes of this discussion a brief description of the connection between the analog and digital systems is in order. Two lines connect the Fundus Reflectometer to the analog-to-digital converter in the computer: a data line and an interrupt line. The data line is a cable over which the varying voltages (i. e., the data) are continuously transmitted for conversion into digital form. The interrupt line enters the converter at such a point that it allows external "pulsing" of the computer; it is used in this installation to signal the computer to start the sampling for a given data value. Upon receipt of an interrupt, the computer is expected to sample the data line as fast as it can for a given number of samples, then find the mean of those samples. The resultant mean is regarded by the program to be the value for one data point. Since the design of the Fundus Reflectometer includes cyclic repetition of values for given wave lengths, and since it is necessary for the computer to be able to differentiate them at any time, it is provided with a signal to indicate the start of a cycle. This signal, called a flag, consists of a

negative voltage on the data line accompanying an interrupt pulse; in other words, a negative data point. A single cycle, consisting of the flag plus all data points before the next flag, is called a scan.

The basic function of FUNDAD is to read scans into memory and put them on tape when time is available to do so.

This basic function is controlled by a variety of parameters which are entered in numeric form via the teletype. The intended location of the data on the cassette is specified as a tape unit number and a file number; the scan is defined in terms of number of points and number of samples per point. The operator then specifies the number of blocks and the scan/skip procedure for each block. A block is defined as an operator-initiated period of data recording; that is to say, the operator must depress the execute button to start a block, since the processor has been halted by the program. For each block, the scan/skip procedure definition consists of the entry of three numbers: the number of times a scan/skip pattern is to be executed within the block; and the pattern itself, consisting of two numbers, the number of scans to be read and the number of scans to be skipped in each execution of the pattern.

When all input parameters have been entered and verified, the program positions the output tape, records the input parameters on it so that it will not be necessary to redescribe the data for each processing program (this record is referred to in other descriptions as the run control information), and halts at the beginning of the first block. When initiated by the

operator, the program executes the scan/skip procedure for that block; upon completion, it halts at the beginning of the second. This procedure is followed until all blocks specified have been recorded; at the end of the last block, the program writes a standard end-of-file on the output tape and transfers control to MONITR to allow the loading of another program.

PROGRAM 1: Voltage printout, short form.

Label: VLTPR1. Entry for first instruction: VLTPR1.

Other entries PROGNO, TAPE. Extns: PRTLIN.

TAPIN, MONITR, FFLOAT, FBDCNV. FADD. FDIV,

BTOD, DTOB, INPUT. BUFIN. Loading order: MONITR,

VLTPR1, FPPKGE, PRTLIN, TAPIN, BTOD, DTOB,

STORES. Storage for mainline subprogram x'ALA'.

This program serves to print the raw voltages for diagnostic and archival purposes. After requesting and obtaining the tape unit and file numbers of the data, the program positions the tape and then performs the following steps:

- (1) Reads the run control information and checks certain values for validity.
- (2) Reads the first block and computes and prints the bias voltage (equal to the mean of all data in the block).

- (3) Reads the second block (baseline), prints each scan in the block ( after subtracting the bias from each value), and computes and prints a pseudo-scan consisting of the mean of all values for each wavelength called the mean baseline scan.
- (4) Reads each record in the third block, and prints all scans in the record ( after subtracting the bias from each value), together with an indication of the pattern that created it.
- (5) When all records in the third block have been processed, returns control to MONITR.

The program is designed to print the data as fast as possible. This is achieved by truncating the actual number of characters printed for each value. Checks are made to ensure that the values are printable in this format; if not, the number is replaced with asterisks in the printout to indicate that it was outside the permissible range. Should this occur, it will be necessary to use the long-form print program (program 2) to obtain the missing values.

PROGRAM 2. Voltage printout, long form.

Label: VLTPR2. Entries, extrns and loading order as for VLTPR1.

Storage for mainline subprograms: x'9D6'.

This program is identical to VLTPR1 (program 1) except that no restriction is made on the size of values printed. It requires approximately fifty percent more printing time than VLTPR1 for a given run.

PROGRAM 3: Voltage averaging without base line.

Label: VLTAV1. Entry for first instruction: VLTPR1.

Other entries: PROGNO, TAPE. Extrns: PRTLIN, TAPIN,

MONITR, FFLOAT, FBDCNV, FADD, FDIV, DTOB, INPUT,

BUFIN. Loading order: MONITR, VLTAV1, FPPKGE, PRTLIN,

TAPIN, DTOB, STORES. Storage for mainline subprogram:

x'844'.

This program computes and prints the mean scan for each scan set in Block 3 of a run, subtracting the bias voltage from each value during the mean computation process. This is accomplished by the following steps:

- (1) As for VLTPR1 Step 1.
- (2) As for VLTPR1 Step 2.
- (3) Skips Block 2 ( baseline data).
- (4) Reads each record in the third block; computes and prints a mean post-flash scan consisting of the mean of all values for each wavelength, for each record in the block.
- (5) When all records in the third block have been processed, returns control to MONITR.

PROGRAM 4: Voltage difference from base line.

Label: VLTAV2. Entry for first instruction: VLTPR1.

Other entries: PROGNO, TAPE. Extrns: PRTLIN, TAPIN,

MONITR, FFLOAT, FBDCNV, FADD, FSUB, FDIV, DTOB,

INPUT, BUFIN. Loading order: MONITR, VLTAV2, FPPKGE,

PRTLIN, TAPIN, DTOB, STORES. Storage for mainline  
subprogram: x'9EA'.

This program computes and prints the difference between the mean post-flash scans as computed by VLTAV1 in Step 4 and the mean baseline scan as computed by VLTPR1 in Step 3. This is accomplished as follows:

- (1) As for VLTPR1 Step 1.
- (2) As for VLTPR1 Step 2.
- (3) Computes and prints the mean baseline scan as in VLTPR1 Step 3. Does not print the individual baseline scans.
- (4) Computes and prints the difference between the mean post-flash scans (computes as in VLTAV1 Step 4) and the mean baseline scan.
- (5) When all records in the third block have been processed, program returns control to MONITR.

PROGRAM 5: Summary tape output from standard run input.

Label: SUMRY1. Entry for first instruction: SUMRY1.

Other entries: PROGNO, TAPE. Extrns: PRTLIN, MONITR, TAPIN, TAPOUT, ENDFIL, FFLOAT, FADD, FDIV, DTOB, INPUT, BUFIN. Loading order: MONITR, SUMRY1, FPPKGE, PRTLIN, TAPIN, TAPOUT, DTOB, STORES. Storage for mainline subprogram: x'A02'.

This program is not a standard processing program. It is designed to accept



multiple standard runs as input, with the further proviso that all runs have the same number of points per scan and the same block patterns. For each run input, the program performs the following steps:

- (1) Requests the location of the run on the data tape.
- (2) Positions the tape accordingly. If this is the first run since the program was initialized, rewinds the output tape and writes a file gap on it.
- (3) Reads the control information of the specified run and checks certain data for validity.
- (4) As for VLTPR1 Step 2 without printing the bias.
- (5) As for VLTAV2 Step 3, except that data is written on tape in hexadecimal floating-point format rather than printed.
- (6) As for VLTAV1 Step 4, except that data is written on tape.
- (7) Writes a sequence number on the output tape indicating the number of standard runs that have thus far been processed onto the summary tape.
- (8) Queries the operator as to whether more runs are to be included. If the operator responds positively, the program returns to (1).

If the operator responds negatively to the question in step 8, the program performs three additional steps:

- (9) Writes an INTERDATA standard end-of-file on the output tape.

(10) Modifies the bootstrap loader (50 sequence) to load the next program on the program tape without using or accessing MONITR.

(11) Exits by entering the 50 sequence to load Program x'7F'.

PROGRAM x'7F': Optical Density Difference calculation.

Label: ODD1. Entry for first instruction: ODD1. Other entries: PROGNO, TAPE. Extns: PRTLIN, TAPIN, MONITR, FFLOAT, FBDCNV, FADD, FSUB, FDIV, FLOG10, BTOD, DTOB. Loading order: MONITR, ODD1, FPPKGE, PRTLIN, TAPIN, BTOD, DTOB, FLOG, POLYEV. Storage for mainline subprogram: x'974'.

This program is not a standard processing program, since it is designed to accept the output summary tape from Program 5 as input. A "run" in the context of this description is meant to indicate the information on the summary tape resulting from the processing of a single standard run by Program 5. Multiple runs may be input, as determined by the input to Program 5. Since the control information written on tape by Program 0 is not passed on to the summary tape by Program 5, it is necessary for the operator to reenter certain control parameters via the teletype. The program is designed to perform the calculation of optical density difference in four possible forms. Given the following definitions:

Let  $V_p$  be a post-flash data value in a mean post-flash scan,

Let  $V_p$  be the value obtained with red filters in that scan,

Let  $V_i$  be a value from either the mean baseline scan or a value from the first mean post-flash scan, and

Let  $V_{ir}$  be the value obtained with red filters in that scan.

We may obtain optical density differences by a choice of two (non-equivalent) formulae:

$$\Delta OD = \text{Log}_{10} ( (V_p - V_{pr}) / (V_i - V_{ir}) ) \quad \text{or}$$

$$\Delta OD = \text{Log}_{10} ( (V_p / V_{pr}) / (V_i / V_{ir}) ).$$

The selection of the type of processing is done by the operator as follows:

Type 1 indicates  $V_i$  is from the baseline and the first formula is to be used;

Type 2 indicates  $V_i$  is from the baseline and the second formula is to be used;

Type 3 indicates  $V_i$  is from the first post-flash scan and the first formula is to be used;

Type 4 indicates  $V_i$  is from the first post-flash scan and the second formula is to be used.

After loading under control of Program 5 and the 50 sequence, the operation of the program proceeds as follows:

- (1) Program corrects 50 sequence to allow proper operation of MONITR;
- (2) Requests number of points per scan, number of repetitions in Block 3 (see abstract for Program 0), and run type from operator;
- (3) Rewinds tape, positions it just before first record, and clears grand mean buffers (referent and result);

- (4) Reads mean baseline scan from tape; on EOF, transfers control to (8), otherwise: if run type is 1 or 2, stores mean baseline scan for future reference, and also adds its values to the grand mean referent buffer; if run type is 3 or 4, the baseline data is ignored;
- (5) Reads first mean post-flash scan from tape; if run type is 3 or 4, scan is treated as baseline and processing continues as in (4) for types 1 and 2; otherwise, the optical density differences are computed using the selected formula, and the results and Vfr added to their proper locations in the grand mean result buffer;
- (6) Each remaining post-flash scan is treated as in (5) for run types 1 and 2;
- (7) Run number of data set just completed is read in, and program transfers to (4).
- (8) Program divides each value in both buffers by the last run number read.
- (9) Program prints the grand mean referent buffer ( the mean of all  $V_i$  and  $V_{ir}$  used in the calculations of optical density differences) and the grand mean result buffer ( the means of all  $V_{fr}$  and optical density differences).
- (10) Program exits via MONITR.

IMPORTANT NOTE: It should be carefully noted that this program does not occupy the position on tape specified by its program number. This is accomplished by the fact that this program is not loadable by the monitor system; rather it is loaded directly by the 50 sequence under direction of Program 5. The program

number is used at the completion of the run to determine if the program is to be reexecuted or a new one loaded, but it is not used at load time. Note also that there is no file gap separating Program 5 and Program x'7F'; the latter is assumed to be part of the former by MONITR.

PROGRAM 6: Tape duplication utility program.

Label: DUPTAP. Entry for first instruction: DUPTAP.

Other entries: PROGNO, TAPE. Extrns: PRTLIN, TAPIN, TAPOUT, ENDFIL, MONITR, DTOB. Loading order: MONITR, DUPTAP, PRTLIN, TAPIN, TAPOUT, DTOB. Storage for mainline subprogram: x'4C2'.

This program is not a standard processing program. Given data written in files on cassette by subroutine TAPOUT, DUPTAP will selectively copy files onto another cassette. Each execution of the program results in a file or sequence of files being copied onto the output tape. The program requests the input and output tape unit numbers, the starting file number for input, the destination file number for the output tape, and the number of files to copy. When the operator answers all these questions via the teletype, the program proceeds automatically, and transfers to MONITR on completion.

The program was primarily written to create backup and archival copies of Fundus Reflectometry data as output by FUNDAD; it will, however, copy any data that is of the format created by subroutine TAPOUT, and as such serves as a general purpose tape duplication program. It will not, however, copy

program tapes: not only does the record size on a program tape (x'1F80') exceed the buffer size of DUPTAP (x'600') but program tape records also are not of the requisite format.

## SUBROUTINE AND ANCILLARY LOAD UNIT ABSTRACTS

The assumption is made that a program calling or referencing one of these subprograms contains the name of the referenced routine in an EXTRN.

MONITR Provides system tape handling interface to the INTERDATA 50 sequence. Entries: MONITR, LOADER. Extrns: PRTLIN. STORAGE: x'100'.

Calling sequence:      B   MONITR to cause typeout of halt  
message and processor  
halt, allowing operator to  
select program in data  
keys; or  
B   LOADER to position tape according  
to selection in keys with-  
out message or halt.

PRTLIN Prints a text buffer on the teletype.  
Entry: PRTLIN. Extrns: none. Storage: x'4C'.  
Calling sequence:      BAL 15,PRTLIN  
DC A(BUFFER),NUMCHR  
where A(BUFFER) is the address  
of the text buffer to be printed, and  
NUMCHR is the halfword integer  
number of characters to be printed.

## **TAPOUT**

Output a buffer to cassette, or write an INTERDATA standard cassette end-of-file. Calling program must contain a halfword labelled TAPE as an entry; this halfword must contain the hardware tape address of the unit on which the selected operation is to be performed.

Entries: TAPOUT, ENDFIL. Extrns: TAPE. Storage: x'1C8'.

Calling sequence (TAPOUT):           BAL 15, TAPOUT  
  DC A(BUFFER)

where BUFFER is a storage area configured as follows:

BUFFER DC n

CKSUM DC 0

DATA DS n

The result of calling the routine in the specified manner is to write a buffer of length n on tape, after computing the checksum and storing it in CKSUM. The data written on tape contains both n and the checksum.

Calling sequence (ENDFIL):           BAL 15, ENDFIL  
which writes a standard INTERDATA end-of-file on the tape unit specified in the halfword labelled TAPE.



## TAPIN

Input a cassette record written by TAPOUT into a buffer. Recompute and verify checksum. Register 15, on return from this routine, contains error indicators as follows:

- 1            Read error;
- 2            Checksum error;
- +1            End-of-file encountered; and
- 0            Data read correctly.

Entry: TAPIN. Extns: TAPE. Storage: x'176'.

Calling sequence:        BAL 15,TAPIN  
                          DC A(BUFFER)  
                          LHR 15,15  
                          BNZ CKOUT

where BUFFER is a storage area configured similarly to that in TAPOUT, except that DATA should be large enough to provide for any expected record, and the halfword constant at BUFFER is altered by TAPIN to indicate the actual number of data bytes read in; and CKOUT is a routine to determine the cause of the non-zero error indicator and take appropriate action.

BTOD

Converts an integer halfword contained in Register 6 to its decimal equivalent in ASCII, storing the generated characters starting at the location specified in Register 11. Leading zeroes are omitted. The number of characters generated, always less than seven, is contained in Register 11 upon return from the routine.

Entry: BTOD. Extns: none. Storage: x'B0'.

Calling sequence:                   LH 6,INTGER  
                                      LHI 11,BUFFER  
                                      BAL 15,BTOD

where INTGER is the halfword to be converted, and BUFFER is the destination address for the character string. Register 6 is unchanged on return from the routine; Register 11 contains the number of bytes generated; and BUFFER contains the generated ASCII string.

DTOB

Converts an ASCII string whose address is contained in Register 11 to a halfword binary integer in Register 6. Leading zeroes and plus sign are optional; the conversion is terminated by a non-numeric ASCII character after the first valid numeric.

Entry: DTOB. Extns: none. Storage: x'E0'.

Calling sequence:                   LHI 11,BUFFER  
                                      BAL 15,DTOB  
                                      STH 6,INTGER

where the character string for conversion is located at BUFFER and the result of the conversion is to be stored at INTGER on return from the routine.

**FLOG10** Computes the common logarithm of a floating-point value contained in Registers 8 and 9, returning the result in the same registers. An error return for zero or negative input values is provided. Entries: FLOGE, FLOG2, FLOG10. Extns: FADD, FMUL, FDIV, FFLOAT (all in FPPKGE), POLYEV. Storage: x'12E'.

Calling sequence:	LH 8,VALUE
	LH 9,VALUE +2
	BAL 15,FLOG10
	B LOGERR
	STH 8,RESULT
	STH 9,RESULT

where LOGERR is a routine to take appropriate action when a negative or zero argument is presented to the routine.

Logarithms to base 2 or e are additionally provided for in this routine; the calling sequence is identical except for the call proper which should be BAL 15,FLOG2 or BAL 15,FLOGE as desired. The appropriate logarithm is computed by the following steps:

- (1) Determine the base to which the logarithm is to be computed by determining the point at which the subroutine was entered.

- (2) Obtain the argument. If negative, take error exit.
- (3) Obtain base-16 exponent from the argument; multiply it by 4 to create a base-2 exponent.
- (4) Obtain fractional part from argument; normalize it to the range  $\frac{1}{2}$  to 1 inclusive by shifting, and for each shift add 1 to the exponent created in (3).
- (5) Convert this new fraction to a standard floating-point number. In the calculations that follow, this value will be referred to as "X".
- (6) Compute the mantissa of the base-2 logarithm of the result of (5), X, using the following expression: <sup>(5)</sup>

$$\frac{((4.81147460989 X + 6.10585199015) X - 8.88626599391) X - 2.05466671951}{((X + 6.42784209029) X + 4.54517087629) X + .353553425277}$$

- (7) Convert the base-2 exponent obtained in (3) and (4) to a floating-point value; add this to the result of (6).
- (8) If the desired base was 2, return with the result of (7) in Registers 8 and 9.
- (9) If the desired base was e, multiply the result of (7) by the natural logarithm of 2, and return with this result in Registers 8 and 9.
- (10) If the desired base was 10, multiply the result of (7) by the common logarithm of 2, and return with this result in Registers 8 and 9.

### POLYEV

Evaluates a polynomial expressed in INTERDATA standard floating-point format.

Entry: POLYEV. Extrns: FADD, FMUL. Storage: x'4A'.

Calling sequence: LHI 2,A1

LHI 4,An

LHI 5,X

BAL 15,POLYEV

Coefficients A1 - An should be in descending order of argument power; for example, the coefficient for  $X^n$ , followed by the  $X^{n-1}$  coefficient, and ending with the constant term. These values must be in consecutive storage locations, and all values must be expressed in floating-point format. After completion of the polynomial evaluation, the result is contained in Registers 8 and 9.

### FPPKGE

INTERDATA standard floating-point package. See INTERDATA publication 29-172 for all information on this set of routines.

### STORES

Storage location EXTRN satisfaction for certain programs in the FRCS. Entries: BUFIN, INADC, INPUT, TOTAP.

Extrns: none. Storage: none. This load unit is not an executable program; in fact, it uses no memory in itself.

It serves as a set of pointers to mainline programs requiring open-ended storage definitions. This program unit is always

loaded after the last executable segment of the program. It defines the buffers requested in relation to the bias supplied by the general loader satisfying the external references from the calling program. All storage from the load bias of this element to the end of memory is assumed to be available for data storage, and any memory extension is automatically made available without program alteration of any sort.